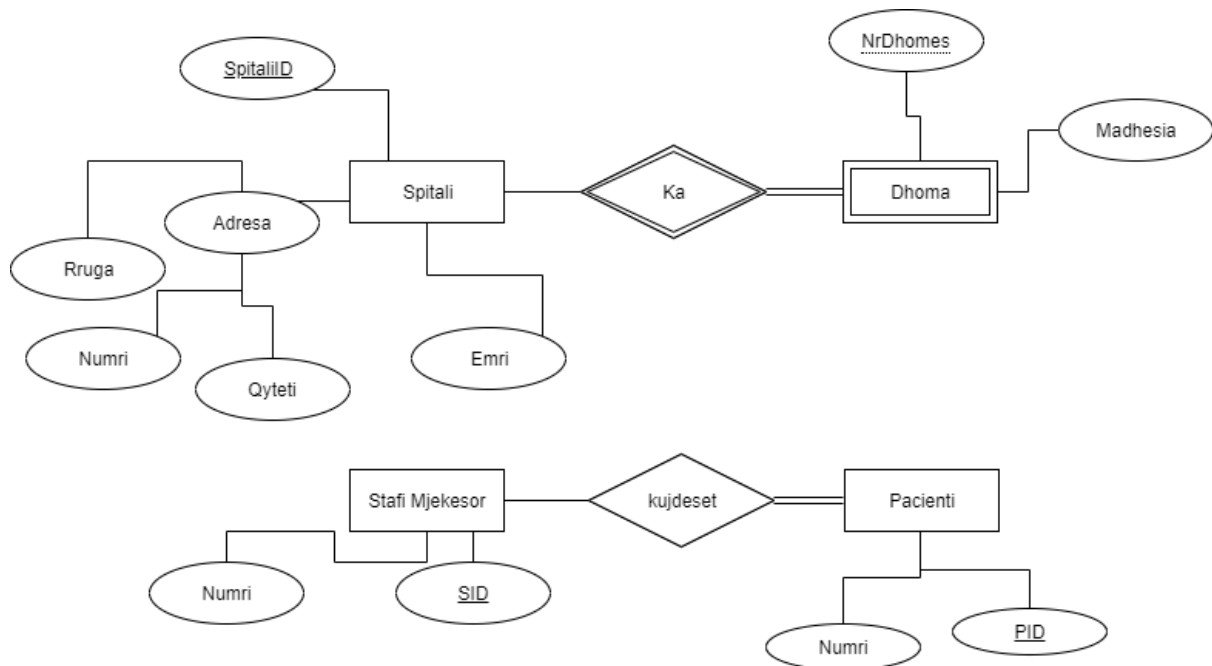


## Zgjidhje për ushtrimet me SQL

1. Krijoni me DDL bazën e të dhënave të quajtur Spitali. Të krijohen tabelat që reprezentojnë dy diagramet e mëposhtme.



```
create database Spitali
use Spitali
```

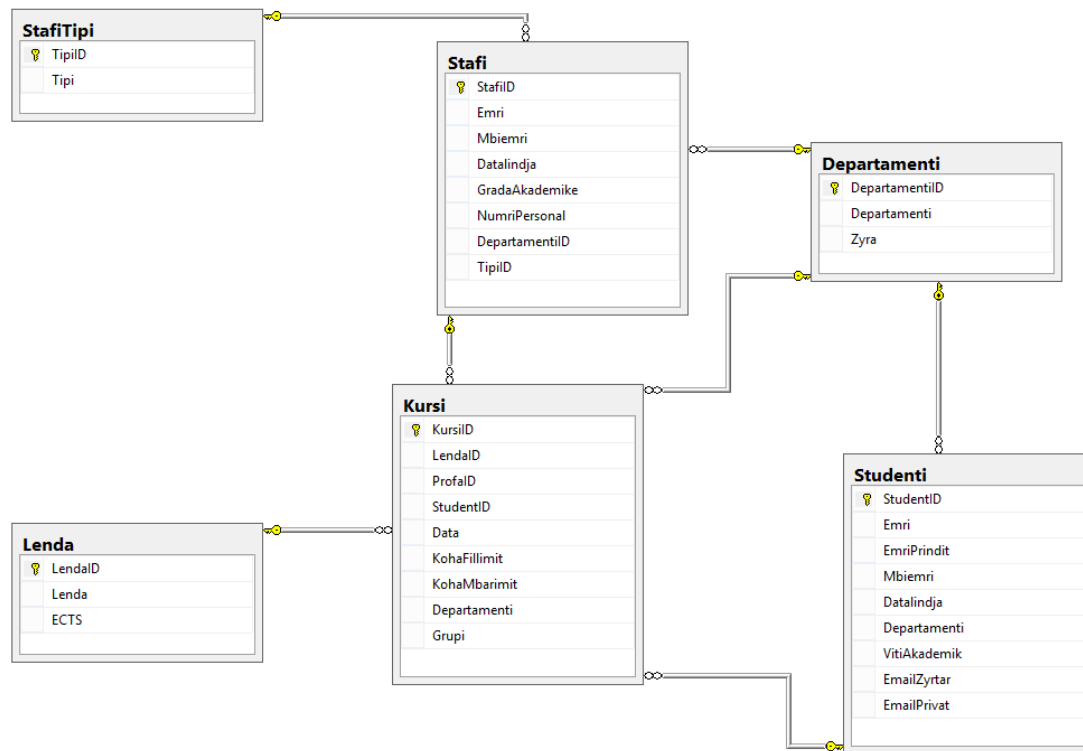
```
create table spitali(
[sid] int primary key ,
emri varchar(255) not null,
rruga varchar(255) not null,
numri int not null,
qyteti char(3) not null
)
```

```
create table dhoma(
nrDhomes int not null,
madhesia int not null,
spitali int foreign key references spitali([sid]),
constraint DhomaID primary key (nrDhomes,spitali)
)
```

```
--rasti per circular dependency
create table punetori(
pid int primary key,
departamenti int
)
create table dep(
depid int primary key,
createdBy int
)
```

```
alter table dep
add foreign key (createdBy) references punetori(pid)
```

```
alter table punetori
add foreign key (departamenti) references dep(depID)
```



```
create database UniversitetiGR2B
use UniversitetiGR2B
```

```
create table stafitipi(
tipiid int primary key,
tipi varchar(255)
)
```

```
create table departamenti(
departamentiid int primary key,
departamenti varchar(255) not null,
zyra int null
)
```

```
create table lenda(
lendaid int primary key,
lenda varchar(255) not null,
ECTS int not null
)
```

```
create table stafi(
stafiid int primary key,
emri varchar(255) not null,
mbiemri varchar(255) not null,
datalindjes date not null,
numriPersonal bigint not null unique,
departamenti int foreign key references departamenti(departamentiid)
)
```

```
alter table stafi
```

```

add tipiid int foreign key references stafitipi(TIPIID)

create table studenti(
studentiid int primary key,
emri varchar(255) not null,
mbiemri varchar(255) not null,
datalindjes date check( datalindjes = '01-03-2020' or
                        datalindjes = '01-11-2020')
)

create table kursi(
kursiid int primary key,
lenda int not null,
profesori int not null,
studenti int not null
)

alter table kursi
add foreign key (lenda) references lenda(lendaaid)

alter table kursi
add foreign key (profesori) references stafi(stafiid)

alter table kursi
add foreign key (studenti) references studenti(studentiid)

```

## trime 8

1. Shfaqni të gjithë personat e qytetit (City) Bronx .

```

select * from person
where city like 'bronx'

```

2. Të shfaqen të gjitha lokacionet (Location) të cilat janë të ruajtura në tabelën OfficeAssignment .

```

select distinct Location
from OfficeAssignment

```

3. Shfaqni te dhënat si: pesha minimale, pesha maksimale, pesha mesatare nga tabela personi (Person).

```

select max(person.[Weight]) as 'maksimumi',min(person.[Weight]) as
'minimumi',avg(person.[Weight]) 'mesatarja'
from person

```

4. Shfaqni emrin, mbiemrin dhe PersonID e personit (Person) te cilën e konvertoni ne varchar duke shtuar parashtesën PERSON.

```
select FirstName, LastName, person.PersonID
from Person
```

5. Shfaqni emrin dhe mbiemrin e personit (Person) duke i bashkuar në një fe të përbashkët “EmriMbiemri” si dhe peshën (Weight) e personit. Njëkohësisht shtoni peshën (Weight) për 3 kg për secilin person (Person) në raport.

```
select FirstName+LastName 'LastName' , person.[weight]+3
from Person
```

6. Gjeni personat e qyteteve (City) Kings, New York, Richmond.

```
select * from person
where city like 'Kings' or city like 'new york' or city like 'richmond'
```

7. Tregoni numrin e personave që jetojnë ne secilin qytet.

```
select person.City, count(*)
from person
group by city
```

8. Shfaqni te gjithë personat (Person) mbiemri (LastName) i te cilëve mbaron me shkronjën „i” dhe peshojnë (Weight) me shume se 76 kg.

```
select *
from person
where LastName like '%i' and person.[Weight]>76
```

9. Tregoi sa persona (Person) kane emrin (FirstName) qe iu fillon me shkronjën F ose A.

```
select *
from person
where person.FirstName like 'F%' or person.FirstName like 'A%'
```

10. Shtoni peshën e Laura Norman për 10%

```
update person set
[Weight]=[weight]*1.1
where FirstName like 'Laura' and LastName like 'Norman'
```

## trime 9

- 1.Shfaqni të gjitha qytetet (**City**) të cilat ndodhen në tabelën **Person**  
(të shfaqet emri i qytetit vetëm një herë në rezultat)

```
select distinct city
from person
```

- 2.Shfaqni të gjitha qytetet (**City**) të cilat ndodhen në tabelën **Person** emri i te cilave përmban shkornjën **n**  
(të shfaqet emri i qytetit vetëm një herë në rezultat dhe të renditen A-Z)

```
select distinct city
from person
where city like '%n%'
```

- 3.Shfaqni të gjitha qytetet (**City**) të cilat ndodhen në tabelën **Person** emri i te cilave përmban shkornjën **n** dhe **r** ose shkornja e dytë e qytetit është **e** (të shfaqet emri i qytetit vetëm një herë në rezultat dhe të renditen Z-A)

```
select distinct city
from person
where city like '_e%' or (city like '%n%' and city like '%r%')
```

- 4.Shfaqni **FirstName** dhe **LastName** të tre personave (**Person**) emri i te cileve fillon me shkronjen **S**

```
select top 3 firstname, LastName
from person
where FirstName like 's%'
```

- 5.Shfaqni **FirstName** dhe **LastName** e të gjith personave (**Person**) të cilet jeton ne qytetët Berlin dhe London (Përdorni **IN**)

6. Shfaqni notën (**Grade**) më të madhe e cila ndodhet në tabelen **StudentGrade**

```
select max(grade)
from StudentGrade
```

7. Shfaqni numrin e notave (**Grade**) 10 të cilat ndodhen në tabelen **StudentGrade**

```
select count(*)
from StudentGrade
where Grade=10
```

8. Shfaqni shumen e të gjitha notave (**Grade**) të cilat ndodhen në tabelen **StudentGrade**

```
select sum(Grade)
from StudentGrade
```

9. Shfaqni të gjithë përsonat (**Person**) të cilët janë studentë dhe janë regjistruar në mes datave 2000-01-01 dhe 2004-12-31

```
select *
from Person
where EnrollmentDate between '2000-01-01' and '2004-12-31'
```

10. Shfaqni të gjitha lëndët (**Course**) dhe ligjeruesit e tyre (**Person**) (**Hint**: përdorni inner join)

```
select FirstName + ' '+LastName, c.Title
from person inner join CourseInstructor ci
on person.PersonID= ci.PersonID
inner join Course c
on c.CourseID=ci.CourseID
```

11. Shfaqni të gjitha lëndët (**Course**) dhe të gjitha departamentet (**Department**) edhe nëse departamenti nuk ka të regjistruar as edhe një lëndë (**Hint**: përdorni right ose left join)

```
select *
from Course right join Department
on Department.DepartmentID=Course.DepartmentID
```

### Detyrë ekstra

Te tregohen lendet, numri i personave qe kane ndjekur atë lende dhe numri i instruktoreve qe kane dhënë atë lende, ku departamenti i kësaj lende është 'Engineering'.

```
select Title, (select count(*)
from StudentGrade sg
where sg.CourseID=c.CourseID) as numriStudenteve,
(select count(*)
from CourseInstructor ci
where ci.CourseID=c.CourseID) as numriLigjeresve
from Course c
```

### Detyrë extra

Te tregohen emri departamentit, emri lendes dhe numri i studentëve, te cilët kane qene te suksesshëm tek ajo lende.

```
select d.Name,c.Title, count(*)
from Course c inner join StudentGrade sg
on sg.CourseID=c.CourseID
inner join Department d
on d.DepartmentID=c.DepartmentID
where sg.Grade>2 -- notat jane nga 1-5
group by title,name
```

## Detyrë extra

Te shfaqen te gjitha kurset qe nuk janë pjese e asnjë departamenti dhe nuk kane asnjë student.

```
select *
from Course c left join StudentGrade sg
                on c.CourseID=sg.CourseID
                left join Department d
                on d.DepartmentID=c.DepartmentID
where sg.CourseID is null and c.DepartmentID is null
```

1. Shfletoni cili shtet sa here është përfaqësuar si fitues më i shpeshtë i ligës se kampionëve.



	Shteti	nr perfaqsimave
1	Spain	6
2	England	3
3	Italy	3
4	Germany	2
5	Portugal	1

```
select Shteti, count(*) as perfaqesimet
from Skuadra s inner join Finalja_LK f
                on s.ID_Skuaadres=f.Kampioni
group by Shteti
order by perfaqesimet desc
```

-- 2. Gjeni për secilën skuadër sa here e ka fituar ligen e kampionëve

	(No column name)	Shteti	Nr i fituesit
1	FC Barcelona	Spain	4
2	FC Real Madrid	Spain	2
3	FC Milan	Italy	2
4	FC Bayem	Germany	2
5	FC Internazionale	Italy	1
6	FC Chelsea	England	1
7	FC Liverpool	England	1
8	FC Man.United	England	1
9	FC Porto	Portugal	1

```
select 'FC ' + Emri, shteti, count(*) perfaqesimet
from Skuadra s inner join Finalja_LK f
                on s.ID_Skuaadres=f.Kampioni
group by emri, Shteti
order by perfaqesimet desc
```

--3. Tregoni për secilën skuadër sa here ka fituar Ligen e kampionëve dhe sa here ka qene nën kampion

	Emri	Kampioni	NenKampioni
1	Barcelona	4	0
2	Juventus	0	2
3	Bayern	2	2
4	Dortmund	0	1
5	Chelsea	1	1
6	Man. United	1	2
7	Internazionale	1	0
8	Milan	2	1
9	Liverpool	1	1
10	Arsenal	0	1
11	Atletico	0	1
12	Real Madrid	2	0
13	Porto	1	0
14	Leverkusen	0	1
15	Valencia	0	1

```
select Emri, (select count(*)
               from Finalja_LK
               where Kampioni=ID_Skuadres) as kampioni,
              (select count(*)
               from Finalja_LK
               where NenKampioni=ID_Skuadres) as nenkampioni
from Skuadra
```

-- 4. Tregoni sa ka qene numri i shikuesve, në të gjitha rastet kur fitues i ligës së kampionëve ishte Barcelona.

	Emri	Shkushmeria
1	Barcelona	318511

```
select s.Emri, sum(st.Kapaciteti_Ulseve) shikueshmeria
from Stadiumi st inner join Finalja_LK
                        on Stadiumi=Stadium_ID
                        inner join Skuadra s
                        on Kampioni=ID_Skuadres
where emri='barcelona'
group by emri
order by shikueshmeria desc
```



-- 5. Shfaqeni numrin e shikueshmërisë për secilin fitues të ligës së kampionëve.

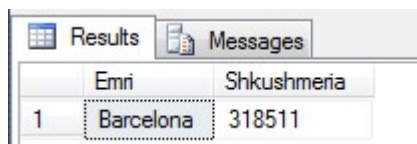


	Emri	Shkushmeria
1	Barcelona	318511
2	Bayern	170018
3	Milan	145271
4	Real Madrid	139250
5	Internazionale	81044
6	Liverpool	80500
7	Porto	78996
8	Man. United	78360
9	Chelsea	75024

```
select s.Emri, sum(st.Kapaciteti_Ulseve) shkushmeria
from Stadiumi st inner join Finalja_LK
                        on Stadiumi=Stadium_ID
                        inner join Skuadra s
                        on Kampioni=ID_Skuadres

group by emri
order by shkushmeria desc
```

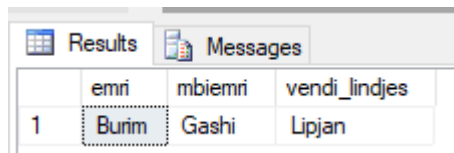
- 6. Gjeni se cili finalist ka pasur shikueshmëri më të madhe prej të gjithë pjesëmarrësve finalist të tjerë.



	Emri	Shkushmeria
1	Barcelona	318511

```
select s.Emri, sum(st.Kapaciteti_Ulseve) shkushmeria
from Stadiumi st inner join Finalja_LK
on Stadiumi=Stadium_ID
inner join Skuadra s
on Kampioni=ID_Skuadres
group by emri
having sum(st.Kapaciteti_Ulseve)>=all (select sum(st.Kapaciteti_Ulseve)
from Stadiumi st inner join Finalja_LK
on Stadiumi=Stadium_ID
inner join Skuadra s
on Kampioni=ID_Skuadres
group by ID_Skuadres)
order by shkushmeria desc
```

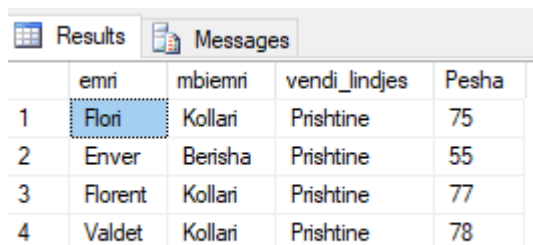
1. Krijo nje view ne te cilin do te selektoni te gjithë Personat te cilet vendin e lindjes e kane 'Lipjan'. Ne view te perfshihet emri, mbiemri dhe vendi\_lindjes



	emri	mbiemri	vendi_lindjes
1	Burim	Gashi	Lipjan

```
create view vendiLindjesLipjan
as
select Emri,Mbiemri,vendi_lindjes
from Personi
where vendi_lindjes='lipjan'
```

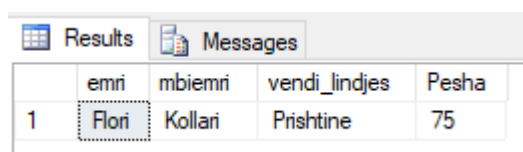
2. Ndryshoni view e me parshem dhe selektoni te gjithë Personat te cilet vendin e lindjes e kane 'Prishtine'. Ne view te perfshihet emri, mbiemri, vendi\_lindjes dhe Pesha.



	emri	mbiemri	vendi_lindjes	Pesha
1	Flori	Kollari	Prishtine	75
2	Enver	Berisha	Prishtine	55
3	Florent	Kollari	Prishtine	77
4	Valdet	Kollari	Prishtine	78

```
alter view vendiLindjesLipjan
as
select Emri,Mbiemri,vendi_lindjes, pesha
from Personi
where vendi_lindjes='prishtine'
```

3. Duke perdorur view me larte shfaq personin qe ka peshen 75kg.



	emri	mbiemri	vendi_lindjes	Pesha
1	Flori	Kollari	Prishtine	75

```
select *
from vendiLindjesLipjan
where Pesha>75
```

4. Krijoni nje view i cili numron te gjithë personat ne baze te vendit te lindjes.

	vendi_lindjes	Persona
1	Ferizaj	1
2	Gjakove	1
3	Lipjan	1
4	Peje	1
5	Prishtine	4
6	Rahovec	2

```
create view numroPersonByVendiLindjes
as
select vendi_lindjes, count(vendi_lindjes)
from Personi
group by vendi_lindjes
```

5. Krijoni nje view qe numron sa Hobi ka secili Person.

	PersoniID	emri	mbiemri	Nr_Hobit
1	1	Burim	Gashi	3
2	4	Kim	Gashi	1
3	5	Flori	Kollari	2
4	7	Yll	Sopa	1
5	9	Ron	Krasniqi	2

```
create view numerHobi as
select PersoniID,Emri,Mbiemri,count(*) as nr_hobit
from Personi p inner join Personi_Hobi ph
on ph.ID_Personi=p.PersoniID
group by PersoniID,Emri,Mbiemri
```

6. Duke perdorur view me larte selekto vetem personat qe kane me shume se 1 Hobi.

	emri	mbiemri	Nr_Hobit
1	Burim	Gashi	3
2	Flori	Kollari	2
3	Ron	Krasniqi	2

```
select emri,Mbiemri,nr_hobit
from numerHobi
where nr_hobit>1
```

Duke perdorur veprimet nga teoria e bashkesive.

1. Të tregohen të gjitha ekipet të cilat nuk kanë shkronjën e parë B.

	ID_Skuadres	Emri	Qyteti	Shteti	Viti_Themelimit
1	2	Juventus	Torino	Italy	1900-01-01
2	4	Dortmund	Dortmund	Germany	1900-01-01
3	5	Chelsea	London	England	1900-01-01
4	6	Man.United	Manchester	England	1900-01-01
5	7	Internazionale	Milano	Italy	1900-01-01
6	8	Milan	Milano	Italy	1900-01-01
7	9	Liverpool	Liverpool	England	1900-01-01
8	10	Arsenal	London	England	1900-01-01
9	11	Atletico	Madrid	Spain	1900-01-01
10	12	Real Madrid	Madrid	Spain	1900-01-01
11	13	Porto	Porto	Portugal	1893-09-28
12	15	Leverkusen	Leverkusen	Germany	1900-01-01
13	16	Valencia	Valencia	Spain	1900-01-01
14	17	Ajax	Amsterdam	Holland	1900-03-18
15	18	Marseille	Marseille	France	1899-01-01
16	19	Sampdoria	Genoa	Italy	1900-01-01
17	20	Crvena zvezda	Belgrade	Serbia	1900-01-01
18	22	Steaua	Bucharest	Romania	1947-06-07
19	23	PSV Eindhoven...	Eindhoven	Holland	1900-01-01
20	24	Roma	Rome	Italy	1900-01-01
21	25	Hamburg	Bucharest	Germany	1887-09-29
22	26	Aston Villa	Birmingham...	England	1900-01-01
23	27	Nottm Forest	Nottingham	England	1900-01-01
24	28	Malmo	Malmo	Sweden	1910-02-24
25	29	Club Brugge	Bruges	Belgium	1891-11-13
26	30	Monchengla...	Monchen...	Germany	1900-01-01
27	31	St-Etienne	Saint-Etie...	France	1900-01-01
28	32	Leeds	Leeds	England	1900-01-01
29	33	Panathinaikos	Athens	Greece	1900-01-01
30	34	Feyenoord	Rotterdam	Holland	1900-01-01
31	35	Celtic	Glasgow	Scotland	1887-11-06
32	36	Partizan	Belgrade	Serbia	1900-01-01
33	37	Frankfurt	Frankfurt	Germany	1900-01-01
34	38	Reims	Reims	France	1900-01-01
35	39	Fiorentina	Florence	Italy	1900-01-01
36	40	Monaco	Monaco	France	1900-01-01

```

use FLK
select *
from Skuadra

except

select *
from Skuadra
where Emri like 'B%'

```

2. Të tregohen të gjitha ekipet të cilat kanë luajtur ne finale.

	Name
1	Arsenal
2	Atletico
3	Barcelona
4	Bayern
5	Chelsea
6	Dortmund
7	Internazionale
8	Juventues
9	Leverkusen
10	Liverpool
11	Man.United
12	Milan
13	Monoco
14	Porto
15	Real Madrid
16	Valencia

```
select emri
from Skuadra inner join Finalja_LK
                        on ID_Skuadres=Kampioni
union
select emri
from Skuadra inner join Finalja_LK
                        on ID_Skuadres=NenKampioni
```

3. Te tregohen të gjitha ekipet që kanë emrin me shkronjën B të paren dhe kanë qenë kampion.

	emri
1	Barcelona
2	Bayern

```
select emri
from Skuadra inner join Finalja_LK
                        on ID_Skuadres=Kampioni
intersect
select emri
from Skuadra
where Emri like 'B%'
```

1. Krijoni një stored procedure për të gjetur personat që kanë peshe më të madhe se pesha mesatare e të gjithë personave.

	emri	mbiemri	Pesha
1	Burim	Gashi	95
2	Florent	Kollari	77
3	Kim	Gashi	85
4	Ron	Krasniqi	90
5	Valdet	Kollari	78

```
create procedure findPersonaByPesha
as
begin
select p.Emri, Mbiemri, Pesha
from Personi p
group by emri, mbiemri, pesha
having pesha > (select avg(pesha)
                from Personi)
end
```

2. Krijoni një stored procedure që shfaq personat në bazë të inputit qyteti.

	PersoniID	Emri	Mbiemri	Data_Lindjes	vendi_lindjes	Ssn	Pesha
1	5	Flori	Kollari	1975-12-23	Prishtine	012-05-1124	75
2	6	Enver	Berisha	1959-10-03	Prishtine	012-05-1125	55
3	8	Florent	Kollari	1987-01-22	Prishtine	012-05-1127	77
4	10	Valdet	Kollari	1989-02-28	Prishtine	012-05-1129	78

```
create proc findPersonsByCity
@city varchar(255)
as
begin
select *
from Personi
where vendi_lindjes=@city
end
```

3. Krijoni një stored procedure në të cilën mund të shfaqen Personat në bazë të inputit ID.

	PersoniID	Emri	Mbiemri	Data_Lindjes	vendi_lindjes	Ssn	Pesha
1	1	Burim	Gashi	1980-11-20	Lipjan	012-05-1120	95

```
create proc findPersonsByID
@personid int
as
begin
select *
from Personi
where PersoniID=@personid
end
```

4. Krijoni një procedure e cila shfaq numrin e hobi-ve te personit ne baze te id-se. Gjeni sa hobi ka personi me id=1 dhe enkripto te dhënat.

	emri	mbiemri	Nr_Hobit
1	Burim	Gashi	3

```
@personiid int
as
begin
select Emri,Mbiemri, count(*) nrHobi
from Personi inner join Personi_Hobi on
personi.PersoniID=Personi_Hobi.ID_Personi
where PersoniID=@personiid
group by emri,Mbiemri
end
```

5. Krijoni një procedure e cila shton një person dhe kete person që sapo krijoi e lidh me hobin “Shah”.

```
create proc addPersonAndAddHoby
@emri varchar(255),
@mbiemri varchar(255),
@datalindjes date,
@vendi varchar(255),
@ssn char(11),
@pesha decimal(18,0)
as
begin
declare @personid int = (select max(p.PersoniID) from Personi p)
insert into Personi values(@personid+1,@emri,@mbiemri,@datalindjes,@vendi,@ssn,@pesha)
declare @id int= (select max(ph.ID_personHobi) from Personi_Hobi ph)
declare @hobiID int = (select ID_hobi from Hobi where hobi.Emri_Hobit='shah')
insert into Personi_Hobi values (@id+1,@personid+1,4)

end
```

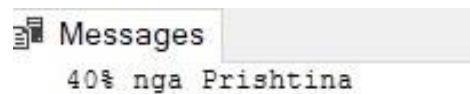
6. Krijoni procedurën e cila gjen personat që nuk kanë asnjë hobi dhe kanë peshën me shumë se inputi që i dërgohet.

```

create proc personNoHoby
@pesha decimal(18,0)
as
begin
select *
from Personi left join Personi_Hobi
on ID_Personi=PersoniID
where Personi_Hobi.ID_Personi is null and Pesha>@pesha
end

```

7. Krijoni një procedure e cila tregon përqindjen e personave nga një qytet, ne baze të inputit të qytetit. Shembull:



```

Messages
40% nga Prishtina

```

```

create proc cityPercentage
@city varchar(255)
as
begin
declare @allPersonCount int = (select count(*) from Personi)
declare @personFromCity int = (select count(*) from Personi where vendi_lindjes=@city)
print 'personat nga ' + @city + ' jane: ' + (@personFromCity/@allPersonCount)*100+ '%'
end

```

8. Krijoni procedure e cila tregon përqindjen e personave që kanë hobi dhe përqindjen e hobive që nuk e ka asnjë person.

```

create proc findPercentage
as
begin
declare @allPersonCount int = (select count(*) from Personi)
declare @PersonCount int = (select count(*) from Personi inner join Personi_Hobi on
ID_Personi=PersoniID)
print 'perqindja e personave qe kane hobi eshte: ' + (@personcount/@allpersoncount)*100
+ '%'

declare @allhobiCount int = (select count(*) from Hobi)
declare @hobiCount int = (select count(*) from hobi left join Personi_Hobi on
hobi.ID_hobi=Personi_Hobi.ID_Hobi)
print 'perqindja e hobive qe nuk kane asnje person eshte: ' +
(@hobicount/@allhobicount)*100 + '%'
end

```